

대한민국 특허청
KOREAN INTELLECTUAL
PROPERTY OFFICE

001/31/02
10/06/2002
U.S. PRO
930/10/06
001/31/02

별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto
is a true copy from the records of the Korean Intellectual
Property Office.

출원번호 : 특허출원 2001년 제 10356 호
Application Number

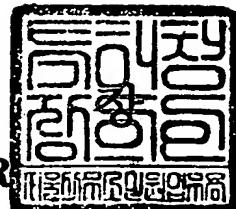
출원년월일 : 2001년 02월 28일
Date of Application

출원인 : 삼성전자 주식회사
Applicant(s)

2001년 04월 06일



특허청
COMMISSIONER



【서류명】	특허출원서		
【권리구분】	특허		
【수신처】	특허청장		
【참조번호】	0004		
【제출일자】	2001.02.28		
【국제특허분류】	H04L		
【발명의 명칭】	채널 활용율을 높이는 통신 시스템 및 그 방법		
【발명의 영문명칭】	Communication system and method for raising coefficient of utilization of channels		
【출원인】			
【명칭】	삼성전자 주식회사		
【출원인코드】	1-1998-104271-3		
【대리인】			
【성명】	이영필		
【대리인코드】	9-1998-000334-6		
【포괄위임등록번호】	1999-009556-9		
【대리인】			
【성명】	정상빈		
【대리인코드】	9-1998-000541-1		
【포괄위임등록번호】	1999-009617-5		
【발명자】			
【성명의 국문표기】	김정현		
【성명의 영문표기】	KIM, Jung Hyun		
【주민등록번호】	730105-1533225		
【우편번호】	449-900		
【주소】	경기도 용인시 기흥읍 농서리 산24번지		
【국적】	KR		
【심사청구】	청구		
【취지】	특허법 제42조의 규정에 의한 출원, 특허법 제60조의 규정에 의한 출원심사 를 청구합니다. 대리인 이영필 (인) 대리인 정상빈 (인)		
【수수료】			
【기본출원료】	20	면	29,000 원
【가산출원료】	10	면	10,000 원

【우선권주장료】	0	건	0	원
【심사청구료】	11	항	461,000	원
【합계】	500,000	원		
【첨부서류】	1.	요약서·명세서(도면)_1통		

【요약서】

【요약】

채널 활용율을 높이는 통신 시스템 및 그 방법이 개시된다. 다수개의 채널들로 이루어진 통신 채널을 통해 수신되는 데이터를 메모리에 저장하거나 또는 메모리에 저장된 팩킷 데이터를 상기 통신 채널을 통해 전송하는 본 발명에 따른 통신 시스템은 각각은, 통신 채널을 통해 수신 또는 송신할 팩킷 데이터에 대한 정보를 저장하고 있는 다수개의 버퍼 디스크립터들, 버퍼 디스크립터들 각각에 팩킷 데이터에 대한 정보를 저장하고, 버퍼 디스크립터들 각각에 표식 비트를 할당하여 버퍼 디스크립터가 구성중인가, 또는 통신 채널로 수신되는 팩킷 데이터에 에러가 발생되었는가 또는 버퍼 디스크립터의 구성이 완료되었는가를 표시하는 CPU(Central Processing Unit) 및 버퍼 디스크립터의 표식 비트를 확인하여 현재 액세스하고자 하는 버퍼 디스크립터의 처리를 중단하고 다음 버퍼 디스크립터를 액세스하거나, 또는 현재 액세스하는 버퍼 디스크립터에 정보가 저장된 팩킷 데이터를 처리하는 DMA(Direct Memory Access) 제어기를 포함함을 특징으로 하고, 다수개의 통신채널을 통해 팩킷 데이터를 송/수신하는 통신 시스템에서, 하나 또는 그 이상의 통신 채널에 오류가 발생되어도 DMA 제어기는 오류가 발생되지 않은 다른 채널로 수신되는 팩킷 데이터를 처리할 수 있어 채널 이용율을 높일 수 있다.

【대표도】

도 3

【명세서】

【발명의 명칭】

채널 활용율을 높이는 통신 시스템 및 그 방법{Communication system and method for raising coefficient of utilization of channels}

【도면의 간단한 설명】

도 1은 CPU에 의해 구성되는 버퍼 디스크립터 구조의 일예를 나타내는 도면이다.

도 2는 CPU에 의해 구현되는 버퍼 디스크립터의 다른 구조를 나타내는 도면이다.

도 3은 본 발명에 따른 채널의 이용율을 높이는 통신 시스템을 개략적으로 나타내는 블록도이다.

도 4는 도 3에 도시된 통신 시스템에서 수행되는 통신 방법의 일실시예에 따른 흐름도이다.

도 5는 도 3에 도시된 DMA 제어기(64)의 다른 실시예를 나타내는 도면이다. 설명의 편의를 위해 도 5에는 버퍼 디스크립터들(66)과 시스템 버스(68)를 함께 도시한다.

도 6은 도 3에 도시된 버퍼 디스크립터(66)의 다른 구성예와, 이를 액세스하는 DMA 제어기 나타내는 도면이다.

【발명의 상세한 설명】

【발명의 목적】

【발명이 속하는 기술분야 및 그 분야의 종래기술】

<7> 본 발명은 통신 시스템에 관한 것으로, 특히, 두 개 이상의 통신 채널들을 통해 데이터를 송/수신하는 통신 시스템에서 통신 채널의 활용율을 높이기 위한 통신 시스템 및

그 방법에 관한 것이다.

<8> 일반적으로, 통신 시스템에서 빠른 데이터 송/수신을 위해 중앙 처리 장치(Central Processing Unit:CPU, 이하, CPU라 칭함)의 간섭없이 직접 메모리 접근(Direct Memory Access:DMA, 이하, DMA라 칭함) 제어기를 이용한다. 즉, DMA 제어기가 통신 채널들을 통해 팩킷(packet) 단위로 수신되는 데이터를 메모리로 저장하거나 또는 메모리에 저장된 데이터를 팩킷 단위로 통신 채널들로 송신한다. 이처럼, DMA 제어기를 이용하여 통신 채널로 수신되는 데이터를 메모리에 저장하거나 또는 메모리에 저장된 데이터를 통신 채널로 송신하는 것은 DMA 제어기가 메모리내의 버퍼 디스크립터(buffer descriptor)를 액세스함으로써 이루어질 수 있다. 버퍼 디스크립터는 CPU가 메모리 내에 통신 채널을 통해 송/수신할 팩킷들에 대한 정보를 미리 구성해놓은 것이다. 여기서, 하나의 버퍼 디스크립터에는 하나의 팩킷에 대한 정보를 저장하고 있다.

<9> DMA 제어기에는 CPU에 의해 설정되는 시작 버퍼 디스크립터 포인터가 있으며, 시작 버퍼 디스크립터 포인터가 지시하는 버퍼 디스크립터를 시작으로 하여 연속되는 버퍼 디스크립터들을 액세스하면서 팩킷 데이터를 처리하게 된다.

<10> 도 1은 CPU에 의해 구성되는 버퍼 디스크립터 구조의 일 예를 나타내는 도면이다.

<11> 도 1을 참조하면, 하나의 버퍼 디스크립터는 데이터 포인터(Data Pointer, DP), 오너 비트(Ownership bit, O), 커맨드 비트(Command, C), 상태 비트(Status, S), 다음 버퍼 디스크립터 포인터(Next Buffer Descriptor Pointer, NBDP)로 이루어진다.

<12> 여기서, 데이터 포인터(DP)는 DMA 제어기가 통신 채널로부터 수신되는 팩킷 데이터가 저장될 메모리의 어드레스 또는 통신 채널로 전송할 팩킷 데이터가 기록되어 있는 메

모리의 어드레스를 나타낸다. 오너 비트(0)는 버퍼 디스크립터가 CPU에 의해 사용 가능한 CPU 모드인가 또는 DMA 제어기에 의해 사용 가능한 DMA 모드인가를 나타내는 비트이다. 커맨드 비트(C)는 팩킷 데이터 처리 커맨드를 나타내는 비트이다. 상태 비트(S)는 팩킷 데이터 송/수신 후 그 결과 상태 및 에러 발생시 에러 유형을 나타내는 비트이다. 다음 버퍼 디스크립터 포인터(NDBP)는 DMA 제어기가 액세스할 다음 버퍼 디스크립터를 지시한다.

<13> 이제, 도 1에 도시된 버퍼 디스크립터를 이용한 DMA 제어기의 팩킷 데이터 처리 방법을 설명한다.

<14> DMA 제어기에는 전술된 바와 같이 CPU에 의해 설정된 버퍼 디스크립터 시작 포인터가 있으며, 설명의 편의를 위해 DMA 제어기의 버퍼 디스크립터 시작 포인터는 제1버퍼 디스크립터를 지시한다고 가정한다. DMA 제어기는 시작 포인터가 지시하는 제1버퍼 디스크립터(10)를 액세스하고, 버퍼 디스크립터(10)의 모드는 오너 비트(0)를 통해 제1버퍼 디스크립터(10)가 CPU 모드인가 또는 DMA 모드인가를 확인한다.

<15> 여기서, 버퍼 디스크립터가 CPU 모드라는 것은 CPU가 데이터 송/수신을 위한 버퍼 디스크립터를 구성 중인 모드이며, 버퍼 디스크립터의 구성이 완료되면 CPU는 오너 비트(0)를 셋시켜 버퍼 디스크립터의 모드를 DMA 모드로 변환한다. DMA 제어기는 제1버퍼 디스크립터(10)의 오너 비트(0)를 확인하여 오너 비트(0)가 셋(또는 리셋)되어 있으면 통신 채널로부터 수신되는 팩킷 데이터를 데이터 포인터(DP)가 지시하는 메모리에 전송한 후, 오너 비트(0)를 리셋시켜 버퍼 디스크립터의 모드를 CPU 모드로 전환시킨다. 이처럼, 버퍼 디스크립터가 CPU 모드가 되면 CPU는 새로운 팩킷에 대한 정보로 버퍼 디스크립터를 구성할 수 있다.

<16> 계속해서, DMA 제어기는 다음 버퍼 디스크립터 포인터(NBDP)를 확인하여 다음 버퍼 디스크립터 포인터가 지시하는 버퍼 디스크립터를 액세스한다. 이 때, 제1버퍼 디스크립터(10)의 다음 버퍼 디스크립터 포인터가 어드레스 '104h'를 지시하면, DMA 제어기는 시작 어드레스가 '104h'인 버퍼 디스크립터(20)를 액세스하여 데이터 처리를 한다.

<17> 만약, DMA 제어기가 액세스하고자 하는 제1버퍼 디스크립터(10)의 오너 비트(0)가 셋되어 있지 않으면 즉, CPU 모드이면 DMA 제어기는 버퍼 디스크립터(10)가 DMA 모드로 될 때까지 기다린다.

<18> 도 2는 CPU에 의해 구현되는 버퍼 디스크립터의 다른 구조를 나타내는 도면이다.

<19> 도 1에 도시된 구조와 비교하면, 도 2에 도시된 버퍼 디스크립터에는 다음 버퍼 디스크립터가 없음을 알 수 있다. 도 2에 도시된 바와 같은 버퍼 디스크립터를 액세스하기 위한 DMA 제어기에는 처음 액세스할 버퍼 디스크립터를 지시하는 버퍼 디스크립터 시작 포인터와 현재 액세스하는 버퍼 디스크립터를 지시하는 현재 버퍼 디스크립터 포인터를 가지고 있다. 그리고, 현재 버퍼 디스크립터 포인터의 초기값은 버퍼 디스크립터 시작 포인터와 같은 값을 가진다.

<20> 도 2를 참조하여, DMA 제어기의 시작 포인터는 100h를 지시한다고 가정하면, DMA 제어기는 제1 버퍼 디스크립터(20)를 액세스하여 먼저 제1버퍼 디스크립터(10)가 CPU 모드인가 또는 DMA 모드인가를 확인한다. DMA 제어기가 제1버퍼 디스크립터(20)에 대한 데이터 처리를 완료하면, DMA 제어기는 현재 포인터의 값에 버퍼 디스크립터의 크기에 해당되는 어드레스를 가산하여 현재 포인터의 값을 바꾼다. 이에 따라, DMA 제어기의 현재 포인터 값은 100h에서 102h로 바뀌게 되며, DMA 제어기는 현재 포인터에 의해 제2버퍼

디스크립터(22)를 액세스한다.

<21> 이처럼, 도 2에 도시된 버퍼 디스크립터의 경우 DMA 제어기는 현재 포인터에 버퍼 디스크립터의 크기만큼의 어드레스를 증가하여 다음에 액세스할 포인터를 알 수 있다. 순차적으로 제n버퍼 디스크립터(28)까지 처리가 완료되면 DMA 제어기의 현재 포인터는 다시 시작 포인터로 되며, 시작 포인터가 지시하는 제1버퍼 디스크립터(20)를 액세스하여 제1버퍼 디스크립터(20)에 의한 데이터 처리를 한다.

<22> 이상에서 설명된 바와 같이, 도 1 및 도 2에 도시된 버퍼 디스크립터 각각에는 DMA 제어기가 사용가능한지의 여부를 나타내는 오너 비트(0)가 있다. DMA 제어기가 오너 비트(0)를 확인한 결과 현재 액세스하고자 하는 버퍼 디스크립터가 CPU 모드이면 DMA 제어기는 버퍼 디스크립터가 DMA 모드로 될 때까지 통신 채널을 통해 송/수신할 팩킷 데이터를 처리하지 못하고 대기하게 된다.

<23> 만약, 두 개 또는 그 이상의 채널을 가지고 팩킷 데이터를 수신하는 통신 시스템에서, 하나의 채널에 오류가 발생하여 수신되는 팩킷 데이터에 오류가 발생되었다면, CPU는 버퍼 디스크립터의 오너 비트를 리셋(또는 셋)시킨다. 이처럼, 버퍼 디스크립터의 오너비트(0)를 리셋(또는 셋)시켜 버퍼 디스크립터의 모드를 CPU 모드로 하면, DMA 제어기는 버퍼 디스크립터를 사용하지 못한다. 따라서, 오류가 발생된 채널에서 수신된 팩킷 데이터가 DMA 제어기를 통해 메모리에 전송되지 않도록 할 수 있다. 이 때, DMA 제어기는 버퍼 디스크립터가 사용 가능해질 때까지 즉, 버퍼 디스크립터가 DMA 모드가 될 때까지 계속해서 대기한다. 이처럼, DMA 제어기가 대기 상태인 동안 오류가 발생되지 않는 다른 채널들을 통해 송신 또는 수신할 팩킷 데이터 역시 처리되지 못한다.

<24> 결국, 두 개 이상의 통신 채널을 통해 팩킷 데이터를 송/수신하는 통신 시스템에서

, 하나의 채널에 오류가 발생되어 DMA 제어기가 대기상태가 되면, DMA 제어기는 오류가 발생되지 않은 다른 채널을 통해 송신 또는 수신할 팩킷 데이터도 처리하지 못하게 된다는 문제점이 있다.

【발명이 이루고자 하는 기술적 과제】

<25> 본 발명이 이루고자 하는 기술적 과제는 다채널 통신 시스템에서 하나 또는 그 이상의 채널에 오류가 발생되어도 DMA 제어기가 오류가 발생되지 않은 다른 채널들을 통해 송/수신되는 팩킷 데이터를 처리할 수 있어 채널 이용율이 높은 통신 시스템을 제공하는 데 있다.

<26> 본 발명이 이루고자 하는 다른 기술적 과제는 상기 통신 시스템에서 이루어지는 통신 방법을 제공하는 데 있다.

【발명의 구성 및 작용】

<27> 상기 과제를 이루기 위해, 다수개의 채널들로 이루어진 통신 채널을 통해 수신되는 데이터를 메모리에 저장하거나 또는 메모리에 저장된 팩킷 데이터를 상기 통신 채널을 통해 전송하는 본 발명에 따른 통신 시스템은 각각은, 통신 채널을 통해 수신 또는 송신 할 팩킷 데이터에 대한 정보를 저장하고 있는 다수개의 버퍼 디스크립터들, 버퍼 디스크립터들 각각에 팩킷 데이터에 대한 정보를 저장하고, 버퍼 디스크립터들 각각에 표식 비트를 할당하여 버퍼 디스크립터가 구성중인가, 또는 통신 채널로 수신되는 팩킷 데이터에 에러가 발생되었는가 또는 버퍼 디스크립터의 구성이 완료되었는가를 표시하는 CPU(Central Processing Unit) 및 버퍼 디스크립터의 표식 비트를 확인하여 현재 액세스하고자 하는 버퍼 디스크립터의 처리를 중단하고 다음 버퍼 디스크립터를 액세스하거나,

또는 현재 액세스하는 버퍼 디스크립터에 정보가 저장된 팩킷 데이터를 처리하는 DMA(Direct Memory Access) 제어기를 포함한다.

<28> 상기 다른 과제를 이루기 위해, CPU가 통신 채널을 통해 수신 또는 송신할 팩킷 데이터에 대한 정보를 저장하고 있는 버퍼 디스크립터를 구성하고, DMA 제어기가 버퍼 디스크립터를 통해 다수개의 통신 채널들을 통해 수신되는 팩킷 데이터를 메모리에 저장하거나 또는 메모리에 저장된 팩킷 데이터를 다수개의 통신 채널들을 통해 전송하는 통신 시스템에서 이루어지는 본 발명에 따른 통신 방법은 CPU는 버퍼 디스크립터를 구성시, 표식 비트를 이용하여 버퍼 디스크립터의 현재 상태를 표시하는 (a)단계, DMA 제어기는 표식 비트를 확인하여 현재 처리하고자 하는 버퍼 디스크립터가 CPU에 의해 구성중인가 또는 통신 채널에 에러가 발생되었는가 또는 버퍼 디스크립터의 구성이 완료되어 DMA 제어기가 사용 가능한 DMA 모드인가를 확인하는 (b)단계, (b)단계에서 현재 액세스하고자 하는 버퍼 디스크립터가 CPU에 의해 구성중이면 버퍼 디스크립터의 구성이 완료될 때까지 표식 비트를 계속적으로 확인하는 (c)단계, (b)단계에서 통신 채널에 에러가 발생된 것으로 확인되면, 현재 액세스하고자 하는 버퍼 디스크립터를 스킵하고 다음 버퍼 디스크립터를 처리하기 위해 상기(b)단계로 진행하는 (d)단계, (b)단계에서 현재 액세스하고 있는 버퍼 디스크립터가 DMA 모드라고 판단되면, DMA 제어기는 액세스하고 있는 버퍼 디스크립터에 정보가 저장된 팩킷 데이터를 처리하는 (e)단계 및 팩킷 데이터의 처리가 완료되면, DMA 제어기는 (e)단계에서 처리한 버퍼 디스크립터를 CPU가 사용할 수 있는 CPU 모드로 변환한 후, 다음 처리할 버퍼 디스크립터로 이동하는 (f)단계로 이루어진다.

<29> 이하, 본 발명에 따른 채널 이용율이 높은 통신 시스템 및 그 방법을 첨부한 도면

들을 참조하여 다음과 같이 설명한다.

<30> 도 3은 본 발명에 따른 채널의 이용률을 높이는 통신 시스템을 개략적으로 나타내는 블록도이다. 본 발명에 따른 통신 시스템은 CPU(60), DMA 제어기(64) 및 다수개의 버퍼 디스크립터들(66)을 포함하여 구성된다. 여기서, 버퍼 디스크립터들(66)은 도시되지는 않았지만 팩킷 데이터가 저장되는 메모리의 일부 영역을 차지한다. 도 3에는 시스템 버스(68)가 함께 도시되어 있으며, CPU(60), DMA 제어기(64) 및 버퍼 디스크립터들(66) 간의 데이터 전송은 시스템 버스(68)를 통해 이루어진다.

<31> 도 3을 참조하여, 버퍼 디스크립터들(66)에는 통신 채널(62)을 통해 팩킷 단위로 수신 또는 송신되는 팩킷 데이터에 대한 정보가 CPU(60)에 의해 저장된다. 그리고, 각 버퍼 디스크립터에는 하나의 팩킷 데이터에 대한 정보가 저장된다. 그리고, 버퍼 디스크립터들(66) 각각은 데이터 포인터(DP), 커맨드 비트(C) 및 상태 비트(S)와, 오너 비트(0) 및 스킵비트(SK)로 이루어진 표식 비트로 구성된다.

<32> CPU(60)는 팩킷 데이터를 저장하는 메모리의 일부영역에 버퍼 디스크립터들(66)을 구성한다. 여기서, CPU(60)가 버퍼 디스크립터를 구성한다는 것은 CPU(60)가 통신 채널(62)을 통해 송/수신하는 팩킷 데이터에 대한 정보를 저장한다는 것이다. 이 때, CPU(60)는 각 버퍼 디스크립터에 표식 비트를 할당하고, 표식비트를 이용하여 버퍼 디스크립터들(66)의 현재 상태를 표시한다. 즉, CPU(60)는 표식비트를 이용하여 버퍼 디스크립터가 CPU(66)에 의해 구성중인가, 또는 통신 채널(62)로 수신되는 팩킷 데이터에 에러가 발생되었는가 또는 버퍼 디스크립터의 구성이 완료되었는가 등의 상태를 표시한다.

<33> CPU(60)는 표식비트로 2비트를 할당하며, 그 중 한 비트는 오너비트(0)로 나머지 한 비트는 스킵비트(SK)로 각각 할당한다.

<34> 오너비트(0)는 버퍼 디스크립터가 CPU 모드인가 또는 DMA 모드인가를 나타낸다. 즉, CPU(60)는 버퍼 디스크립터를 구성중이거나 또는 통신 채널(62)로 수신되는 팩킷 데이터에 에러가 발생된 경우, 오너비트(0)를 셋(또는 리셋, 이하, 셋인 것으로 가정한다)하여 버퍼 디스크립터가 현재 CPU 모드임을 나타낸다. 이처럼, 버퍼 디스크립터가 CPU 모드이면 DMA 제어기(64)는 버퍼 디스크립터를 사용하지 못한다. 또한, CPU(60)는 버퍼 디스크립터의 구성이 완료되면 오너비트(0)를 리셋(또는 셋, 이하, 리셋인 것으로 가정한다)하여 버퍼 디스크립터가 DMA 제어기(64)에 의해 사용 가능한 DMA 모드임을 나타낸다.

<35> 스킵 비트(SK)는 CPU(60)가 버퍼 디스크립터를 구성중인가 또는 통신 채널(62)로 수신되는 팩킷 데이터에 에러가 발생되었는가를 표시한다. 즉, CPU(60)는 버퍼 디스크립터를 구성중이면 스킵 비트(SK)를 리셋(또는 셋, 이하 리셋인 것으로 가정한다)한다. 반면, 통신 채널(62)에 오류가 발생되어 통신 채널(62)로 수신되는 패킷에 오류가 발생되면, CPU(60)는 스kip 비트(SK)를 셋(또는 리셋, 이하, 리셋인 것으로 가정한다)한다.

<36> DMA 제어기(64)는 버퍼 디스크립터의 시작 포인터와 현재 포인터를 가지고 있다. 시작 포인터(64a)는 CPU(60)에 의해 설정되며, DMA 제어기(64)가 처음으로 액세스하는 버퍼 디스크립터를 지시한다. 현재 포인터(64b)는 DMA 제어기(64)가 현재 액세스하고 있는 버퍼 디스크립터를 지시하며, 초기적으로 시작 포인터와 같은 값을 가진다. 현재 포인터(64b)는 이전에 처리된 버퍼 디스크립터의 포인터 값에 단위 버퍼 디스크립터의 크기에 해당되는 어드레스를 증가하여 그 값을 얻는다. 따라서, DMA 제어기(64)의 현재 포인터(64b)의 같은 처음에는 시작 포인터(64a)의 값과 같은 제1버퍼 디스크립터(70)를 지시하며, 순차적으로 제2, 제3, ..., 제n버퍼 디스크립터(72, 74, ..., 78)를 지시한다.

DMA 제어기(64)는 현재 포인터(64b)에 따라 제1, 제2, 제3, ..., 제n버퍼 디스크립터(70, 72, 74, ..., 78)를 순차적으로 처리한다.

<37> DMA 제어기(64)는 버퍼 디스크립터들(66) 각각의 표식 비트를 확인하여, 현재 액세스하고 있는 버퍼 디스크립터의 처리를 중단하고 다음 버퍼 디스크립터를 액세스하거나, 또는 현재 액세스하고 있는 버퍼 디스크립터에 저장된 정보를 이용하여 통신 채널(62)을 통해 송/수신할 팩킷 데이터를 처리한다. 여기서, 팩킷 데이터 처리는 버퍼 디스크립터를 통해 통신 채널(62)로부터 수신되는 팩킷 데이터를 메모리로 전송하거나 또는 메모리에 저장된 팩킷 데이터를 통신 채널(62)로 전송한다는 것이다.

<38> 도 4는 도 3에 도시된 통신 시스템에서 수행되는 통신 방법의 일실시예에 따른 흐름도이다.

<39> 이제, 도 3 내지 도 4를 참조하여 도 3에 도시된 시스템에서 수행되는 본 발명에 따른 통신 방법을 설명한다.

<40> 도 3 및 도 4를 참조하여, CPU(60)는 n개의 채널로 이루어진 통신 채널(62)로 송/수신되는 팩킷 데이터에 대한 정보를 버퍼 디스크립터들(66)에 저장한다(제98단계). 이 때, 하나의 버퍼 디스크립터에는 하나의 팩킷 데이터에 대한 정보가 저장되며, CPU(60)는 각 버퍼 디스크립터에는 버퍼 디스크립터의 상태를 표시하는 2비트의 표식비트를 할당한다. CPU(60)는 표식 비트를 이용하여, 버퍼 디스크립터에 팩킷 데이터에 대한 정보를 저장중인가, 현재 저장중인 팩킷 데이터를 수신한 통신 채널(62)에 오류가 발생되었는가 또는 팩킷 데이터에 대한 정보 저장이 완료되었는가를 나타낸다.

<41> 제98단계 후에, DMA 제어기(64)는 현재 포인터(64b)가 지시하는 버퍼 디스크립터를

액세스한다(제100단계). 설명의 편의를 위해, 현재 포인터(64b)는 제i버퍼 디스크립터(76)를 지시하고 있는 것으로 한다.

<42> DMA 제어기(64)는 액세스하고자 하는 제i버퍼 디스크립터(76)의 표식비트를 확인하여 현재 액세스하고 있는 제i버퍼 디스크립터(76)의 현재 상태를 확인한다(제175단계). 즉, DMA 제어기(64)는 표식비트를 확인하여, 제i버퍼 디스크립터(76)에 팩킷 데이터에 대한 정보가 저장되는 중인가, 팩킷 데이터를 송/수신하는 통신 채널에 오류가 발생되었는가 또는 팩킷 데이터에 대한 정보 저장이 완료되었는가를 확인한다.

<43> 구체적으로, DMA 제어기(64)는 표식 비트의 오너 비트(0)를 확인하여 현재 액세스하고 있는 제i버퍼 디스크립터(76)의 모드가 CPU 모드인가 또는 팩킷 데이터에 대한 정보 저장이 완료된 DMA 모드인가를 판단한다(제110단계). 예컨대, 오너 비트(0)가 셋되어 있으면 DMA 제어기(64)는 제i버퍼 디스크립터(76)가 CPU 모드인 것으로 판단하고, 리셋되어 있으면 DMA 모드인 것으로 판단한다.

<44> 제110단계에서, 제i버퍼 디스크립터(76)가 CPU 모드라고 판단되면, 스킵비트(SK)를 확인하여 통신 채널(62)로 수신되는 팩킷 데이터에 대한 정보가 제i버퍼 디스크립터(76)에 저장되는 중인가 또는 팩킷 데이터를 수신하는 통신 채널(62)에 오류가 발생되었는가를 확인한다(제150단계). 예컨대, DMA 제어기(64)는 스kip 비트(SK)가 셋되어 있으면 통신 채널(62)에 오류가 발생된 것으로 판단하고, 리셋되어 있으면 통신 채널(62)로 수신되는 팩킷 데이터에 대한 정보가 제i버퍼 디스크립터(76)에 저장되는 중인 것으로 판단한다.

<45> 계속해서, 제175단계에서 DMA 제어기(64)가 액세스하고 있는 제i버퍼 디스크립터(76)가 DMA 모드라고 확인되면, 제i버퍼 디스크립터(76)에 정보가 저장된 팩킷 데이터에

대한 처리를 한다(제120단계). 여기서, 팩킷 데이터를 처리한다는 것은, DMA 제어기(64)가 제i버퍼 디스크립터(76)의 데이터 포인터(DP)가 지시하는 메모리의 어드레스로 팩킷 데이터를 저장하거나 또는 데이터 포인터(DP)가 지시하는 메모리의 어드레스에 저장된 팩킷 데이터를 통신 채널(62)로 전송한다는 것이다.

<46> 제i버퍼 디스크립터(76)에 정보가 저장된 팩킷 데이터의 처리가 완료되면, DMA 제어기(64)는 제i버퍼 디스크립터(76)의 모드를 CPU 모드로 변환하고, 다음에 처리할 버퍼 디스크립터를 액세스한다(제130단계). 즉, DMA 제어기(64)는 제i버퍼 디스크립터(76)에 정보가 저장된 팩킷 데이터의 처리가 완료되면, 표식 비트의 오너 비트(0)를 셋시켜, 제i버퍼 디스크립터(76)의 모드를 CPU 모드로 변환한다. 이처럼, 오너 비트(0)가 셋되면 CPU(60)는 제i버퍼 디스크립터(76)에 새로운 팩킷 데이터에 대한 정보를 저장한다. 또한, 제i버퍼 디스크립터(76)에 정보가 저장된 팩킷 데이터의 처리가 완료되면, DMA 제어기(64)의 현재 포인터(64b)는 제(i+1) 버퍼 디스크립터를 지시한다. DMA 제어기(64)는 현재 포인터(64b)를 참조하여 제(i+1)버퍼 디스크립터를 액세스한다.

<47> 한편, 제175단계에서 CPU(60)가 통신 채널(62)중 하나의 채널로 송/수신할 팩킷 데이터에 대한 정보를 제i버퍼 디스크립터(76)에 저장중이라고 판단되면, DMA 제어기(64)는 버퍼 디스크립터에 팩킷 데이터 정보에 대한 저장이 완료될 때까지 대기한다(제160단계).

<48> 반면, 제175단계에서 현재 그 정보를 저장중인 팩킷 데이터에 오류가 발생되었다고 판단되면 DMA 제어기(64)는 현재 액세스하고 있는 제i버퍼 디스크립터(76)의 처리를 중단하고 다음 버퍼 디스크립터인 제(i+1)버퍼 디스크립터를 액세스한다(제170단계). 즉, DMA 제어기(64)는 스킵 비트(SK)를 확인하여 현재 처리하고자 하는 팩킷 데이터를 송/수

신하는 통신 채널(62)에 오류가 발생되었다면, 현재 액세스하고 있는 버퍼 디스크립터의 처리를 중단한다. 그리고, 오류가 발생되지 않은 통신 채널로 송/수신하는 팩킷 데이터에 대한 정보를 저장하고 있는 다른 버퍼 디스크립터를 액세스하여 팩킷 데이터를 처리한다.

<49> 이상에서와 같이, 본 발명에 따른 통신 시스템에서는 CPU(60)가 버퍼 디스크립터를 구성시 통신 채널에 오류 발생 여부를 나타내는 스kip 비트(SK)를 할당한다. DMA 제어기(64)는 이 스kip 비트(SK)를 통해 처리하고자 하는 팩킷 데이터를 송/수신하는 통신 채널에 오류가 발생되었는가의 여부를 확인할 수 있다. 즉, DMA 제어기(64)는, 처리하고자 하는 팩킷 데이터를 송/수신하는 통신 채널에 오류가 발생되었다면, 오류가 발생되지 않은 다른 통신 채널을 통해 송/수신할 팩킷 데이터를 먼저 처리함으로써, 통신 채널의 이용율을 효과적으로 높일 수 있다.

<50> 도 5는 도 3에 도시된 DMA 제어기(64)의 다른 실시예를 나타내는 도면이다. 설명의 편의를 위해 도 5에는 버퍼 디스크립터들(66)과 시스템 버스(68)를 함께 도시한다.

<51> DMA 제어기(200)는 시작 포인터(200a)와 어드레스 카운터(200b)를 구비한다. 시작 포인터(200a)는 CPU(60)에 의해 설정되며, DMA 제어기(64)가 처음으로 액세스하는 버퍼 디스크립터를 지시한다. 그리고, 어드레스 카운터(200b)는 DMA 제어기(200)가 현재 액세스하고 있는 버퍼 디스크립터의 시작 어드레스를 카운트하며, 시작 포인터(200a)가 지시하는 값으로 초기화된다. 즉, 어드레스 카운터(200b)는 이전에 처리된 버퍼 디스크립터의 시작 어드레스에 단위 버퍼 디스크립터의 크기에 해당되는 어드레스를 가산하여 다음에 액세스할 버퍼 디스크립터의 시작 어드레스를 얻는다.

<52> 구체적으로, DMA 제어기(200)는 시작 포인터(200a)가 지시하는 제1버퍼 디스크립터(80)를 처음 액세스한다. 예를 들어, 시작 포인터(200a)가 100h를 지시 한다고 하면, DMA 제어기(200)는 어드레스 100h를 액세스하여 팩킷 데이터 처리를 한다. 제1 버퍼 디스크립터(80)에 대한 팩킷 데이터의 처리가 완료되면, 어드레스 카운터(200b)는 현재 카운터 값에 단위 버퍼 디스크립터의 크기에 해당되는 어드레스(예컨대, 004h)를 가산하여 다음에 액세스할 버퍼 디스크립터의 시작 어드레스 104h를 카운트한다. DMA 제어기(200)는 어드레스 카운터(200b)를 참조하여 제2버퍼 디스크립터(82)를 액세스한다.

<53> 이처럼, DMA 제어기(200)가 어드레스 카운터(200b)를 이용하여 다음에 액세스할 버퍼 디스크립터의 시작 어드레스를 얻는 경우, DMA 제어기(200)가 소프트웨어적으로 관리해야 하는 포인터가 시작 포인터 하나이므로 포인터 처리를 위한 프로그래밍이 간단해질 수 있다.

<54> DMA 제어기(200)가 버퍼 디스크립터들(66)을 이용하여 팩킷 데이터를 처리하는 방법은 도 4를 참조하여 설명된 것과 같으므로, 여기서는 그 상세한 설명을 생략한다.

<55> 도 6은 도 3에 도시된 버퍼 디스크립터(66)의 다른 구성예와, 이를 액세스하는 DMA 제어기 나타내는 도면이다.

<56> 도 3 및 도 6을 참조하여, 하나의 버퍼 디스크립터는 데이터 포인터(DP), 커맨드 비트(C), 상태 비트(S), 다음 버퍼 디스크립터 포인터(NBDP)와, 오너 비트(O) 및 스킵 비트(SK)로 이루어진 표식 비트로 이루어진다. 여기서, 다음 버퍼 디스크립터 포인터(NBDP)는 DMA 제어기(64)가 액세스할 다음 버퍼 디스크립터를 지시한다. 즉, 각 버퍼 디스크립터에 다음 버퍼 디스크립터 포인터를 구비하고 있으므로, DMA 제어기(300)에는 처음 액세스할 버퍼 디스크립터를 지시하는 시작 포인터(300a)만 있으면 되며, 이는

CPU(60)에 의해 설정된다.

<57> DMA 제어기(200)가 버퍼 디스크립터들(66)을 이용하여 팩킷 데이터를 처리하는 방법은 도 4를 참조하여 설명된 것과 같으므로, 여기서는 그 상세한 설명을 생략한다.

<58> 다음 표 1은 오류 확률이 각각 5%, 10%인 통신 채널에서 연속적인 프레임 수신 확률을 나타낸다.

<59> 【표 1】

연속 프레임 수	오류 확률 5%인 통신 채널의 수신 확률	오류 확률 10%인 통신 채널의 수신 확률
10	60%	35%
20	36%	12.3%
50	8%	0.5%
100	0.64%	0.003%

<60> 표 1을 참조하면, 통신 채널에서 오류 확률이 5% 또는 10%인 경우 연속해서 수신할 수 있는 프레임의 수가 많아질수록 연속된 프레임을 정상적으로 수신할 확률이 매우 작아진다. 이는 통신 채널을 통해 수신되는 팩킷 데이터에 오류가 발생될 확률이 그만큼 높다는 것을 나타낸다.

<61> 다음 표 2는 10프레임을 연속적으로 수신하고, 각 통신 채널이 5%의 오류 확률이 있을 경우, 종래 기술과 본 발명에 따른 채널 이용율을 나타낸다.

<62> 【표 2】

채널 수	채널 이용률(종래)	채널 이용률(본 발명)
2	36%	84%
3	22%	93%
5	8%	99%
10	0.64%	100%

<63> 표 2를 참조하면, 종래에는 채널 수가 많아도 한 채널에 오류가 발생되면 오류가

발생된 채널로 인해 오류가 발생되지 않은 다른 채널들 모두를 이용할 수 없다. 따라서, 채널 수가 많아질수록 채널 이용율이 급격히 떨어진다. 그러나, 본 발명에서는 하나 또는 그 이상의 통신 채널에 오류가 발생되어도 오류가 발생되지 않은 다른 채널로 수신되는 팩킷 데이터를 처리할 수 있음으로써, 채널 수가 많아질수록 채널 이용율을 높일 수 있다.

<64> 이상 도면과 명세서에서 최적 실시예들이 개시되었다. 여기서 특정한 용어들이 사용되었으나, 이는 단지 본 발명을 설명하기 위한 목적에서 사용된 것이지 의미 한정이나 특허청구범위에 기재된 본 발명의 범위를 제한하기 위하여 사용된 것은 아니다. 그러므로 본 기술 분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시 예가 가능하다는 점을 이해할 것이다. 따라서, 본 발명의 진정한 기술적 보호 범위는 첨부된 특허청구범위의 기술적 사상에 의해 정해져야 할 것이다.

【발명의 효과】

<65> 상술한 바와 같이, 본 발명에 따른 채널 이용율을 높이는 통신 시스템 및 그 방법에 따르면, 다수개의 통신채널을 통해 팩킷 데이터를 송/수신하는 통신 시스템에서, 하나 또는 그 이상의 통신 채널에 오류가 발생되어도 DMA 제어기는 오류가 발생되지 않은 다른 채널로 수신되는 팩킷 데이터를 처리할 수 있어 채널 이용율을 높일 수 있다.

【특허청구범위】**【청구항 1】**

다수개의 채널들로 이루어진 통신 채널을 통해 수신되는 데이터를 메모리에 저장하거나 또는 메모리에 저장된 팩킷 데이터를 상기 통신 채널을 통해 전송하는 통신 시스템에 있어서,

각각은, 상기 통신 채널을 통해 수신 또는 송신할 팩킷 데이터에 대한 정보를 저장하고 있는 다수개의 버퍼 디스크립터들;

상기 버퍼 디스크립터들 각각에 상기 팩킷 데이터에 대한 정보를 저장하고, 상기 버퍼 디스크립터들 각각에 표식 비트를 할당하여 상기 버퍼 디스크립터가 구성중인가, 또는 상기 통신 채널로 수신되는 팩킷 데이터에 에러가 발생되었는가 또는 상기 버퍼 디스크립터의 구성이 완료되었는가를 표시하는 CPU(Central Processing Unit); 및

상기 버퍼 디스크립터의 표식 비트를 확인하여 현재 액세스하고자 하는 버퍼 디스크립터의 처리를 중단하고 다음 버퍼 디스크립터를 액세스하거나, 또는 현재 액세스하는 버퍼 디스크립터에 정보가 저장된 팩킷 데이터를 처리하는 DMA(Direct Memory Access) 제어 제어기를 포함하는 것을 특징으로 하는 통신 시스템.

【청구항 2】

제1항에 있어서, 상기 표식 비트는

상기 CPU가 상기 버퍼 디스크립터를 구성중이거나 또는 상기 통신 채널로 수신되는 팩킷 데이터에 에러가 발생되면 상기 버퍼 디스크립터가 CPU 모드이고, 상기 CPU에 의해 상기 버퍼 디스크립터의 구성이 완료되면 상기 DMA 제어기가 사용 가능한 DMA 모드

임을 나타내는 오너 비트; 및

상기 CPU가 상기 버퍼 디스크립터를 구성중인가 또는 상기 통신 채널로 수신되는 팩킷 데이터에 에러가 발생되었는가를 표시하는 스킵 비트로 이루어지는 것을 특징으로 하는 통신 시스템.

【청구항 3】

제1항에 있어서,

상기 CPU는 상기 버퍼 디스크립터에 다음 버퍼 디스크립터 포인터를 할당하고, 상기 DMA 제어기는 상기 버퍼 디스크립터에 구성되는 다음 버퍼 디스크립터 포인터를 확인하여 다음에 처리할 버퍼 디스크립터를 액세스하는 것을 특징으로 하는 통신 시스템.

【청구항 4】

제1항에 있어서,

상기 DMA 제어기는 현재 액세스하고 있는 버퍼 디스크립터의 처리가 완료되면, 현재 액세스하고 있는 버퍼 디스크립터를 지시하는 현재 포인터의 값에 상기 버퍼 디스크립터의 크기에 해당되는 어드레스를 가산하여 현재 포인터값을 갱신하는 것을 특징으로 하는 통신 시스템.

【청구항 5】

제1항에 있어서, 상기 DMA 제어기는

처음 액세스할 버퍼 디스크립터를 지시하는 시작 포인터; 및

상기 시작 포인터의 값으로 초기화되고, 상기 버퍼 디스크립터의 시작 어드레스를 카운팅하는 어드레스 카운터를 구비하고,

상기 어드레스 카운터는 현재 어드레스 카운터의 값에 단위 버퍼 디스크립터의 크기에 해당되는 어드레스를 가산하여 현재 어드레스 카운터의 값을 갱신하는 것을 특징으로 하는 통신 시스템.

【청구항 6】

CPU가 통신 채널을 통해 수신 또는 송신할 팩킷 데이터에 대한 정보를 저장하고 있는 버퍼 디스크립터를 구성하고, DMA 제어기가 상기 버퍼 디스크립터를 통해 다수개의 통신 채널들을 통해 수신되는 팩킷 데이터를 메모리에 저장하거나 또는 메모리에 저장된 팩킷 데이터를 상기 다수개의 통신 채널들을 통해 전송하는 통신 시스템에서 이루어지는 통신 방법에 있어서,

- (a) 상기 CPU는 상기 버퍼 디스크립터를 구성시, 표식 비트를 이용하여 상기 버퍼 디스크립터의 현재 상태를 표시하는 단계;
- (b) DMA 제어기는 상기 표식 비트를 확인하여, 현재 처리하고자 하는 버퍼 디스크립터가 상기 CPU에 의해 구성중인가 또는 상기 통신 채널에 에러가 발생되었는가 또는 상기 버퍼 디스크립터의 구성이 완료되어 상기 DMA 제어기가 사용 가능한 DMA 모드인가를 확인하는 단계;
- (c) 상기 (b) 단계에서 현재 액세스하고자 하는 버퍼 디스크립터가 상기 CPU에 의해 구성중이면 버퍼 디스크립터의 구성이 완료될 때까지 상기 표식 비트를 계속적으로 확인하는 단계;
- (d) 상기 (b) 단계에서 상기 통신 채널에 에러가 발생된 것으로 확인되면, 현재 액세스하고자 하는 버퍼 디스크립터를 스kip하고 다음 버퍼 디스크립터를 처리하기 위해

상기(b)단계로 진행하는 단계;

(e) 상기 (b)단계에서 현재 액세스하고 있는 버퍼 디스크립터가 DMA 모드라고 판단되면, 상기 DMA 제어기는 액세스하고 있는 버퍼 디스크립터에 정보가 저장된 팩킷 데이터를 처리하는 단계; 및

(f) 팩킷 데이터의 처리가 완료되면, 상기 DMA 제어기는 상기 (e)단계에서 처리한 버퍼 디스크립터를 CPU가 사용할 수 있는 CPU 모드로 변환한 후, 다음 처리할 버퍼 디스크립터로 이동하는 단계로 이루어지는 것을 특징으로 하는 통신 방법.

【청구항 7】

제6항에 있어서, 상기 표식 비트는

상기 CPU가 상기 버퍼 디스크립터를 구성중이거나 또는 상기 통신 채널로 수신되는 팩킷 데이터에 에러가 발생되면 상기 버퍼 디스크립터가 상기 CPU 모드이고, 상기 CPU에 의해 상기 버퍼 디스크립터의 구성이 완료되면 상기 DMA 모드임을 나타내는 오너 비트; 및

상기 CPU가 상기 버퍼 디스크립터를 구성중인가 또는 상기 통신 채널로 수신되는 팩킷 데이터에 에러가 발생되었는가를 표시하는 스킵 비트로 이루어지는 것을 특징으로 하는 통신 방법.

【청구항 8】

제7항에 있어서, 상기 (b)단계는

(b1) 상기 DMA 제어기는 상기 오너 비트를 통해 현재 처리하고자 하는 버퍼 디스크립터의 모드가 상기 CPU 모드인가 또는 상기 DMA 모드인가를 확인하는 단계;

(b2)상기 (b1)단계에서 처리하고자 하는 버퍼 디스크립터의 모드가 상기 DMA 모드라고 확인되면 상기 (e)단계로 진행하고, 처리하고자 하는 버퍼 디스크립터의 모드가 상기 CPU 모드라고 확인되면 상기 스킵 비트를 확인하는 단계; 및

(b3)상기 스kip 비트를 확인하여, 상기 통신 채널에 에러가 발생된 것이 확인되면 상기 (d)단계로 진행하고, 상기 CPU에 의해 구성중인 것으로 확인되면 상기 (c)단계로 진행하는 것을 특징으로 하는 통신 방법.

【청구항 9】

제6항에 있어서,

상기 CPU는 상기 버퍼 디스크립터에 다음 버퍼 디스크립터 포인터를 할당하고, 상기 DMA 제어기는 상기 버퍼 디스크립터에 구성되는 다음 버퍼 디스크립터 포인터를 확인하여 다음에 처리할 버퍼 디스크립터를 액세스하는 것을 특징으로 하는 통신 시스템.

【청구항 10】

제6항에 있어서,

상기 DMA 제어기는 현재 액세스하고 있는 버퍼 디스크립터의 처리가 완료되면, 현재 액세스하고 있는 버퍼 디스크립터를 지시하는 현재 포인터의 값에 상기 버퍼 디스크립터의 크기에 해당되는 어드레스를 가산하여 현재 포인터값을 갱신하는 것을 특징으로 하는 통신 시스템.

【청구항 11】

제1항에 있어서, 상기 DMA 제어기는

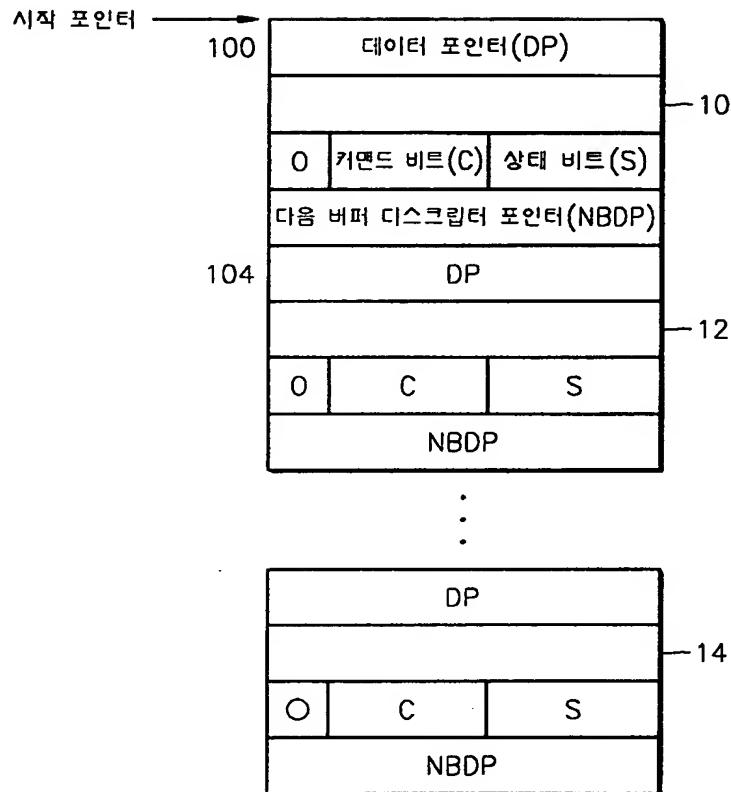
처음 액세스할 버퍼 디스크립터를 지시하는 시작 포인터; 및

상기 시작 포인터의 값으로 초기화되고, 상기 버퍼 디스크립터의 시작 어드레스를 카운팅하는 어드레스 카운터를 구비하고,

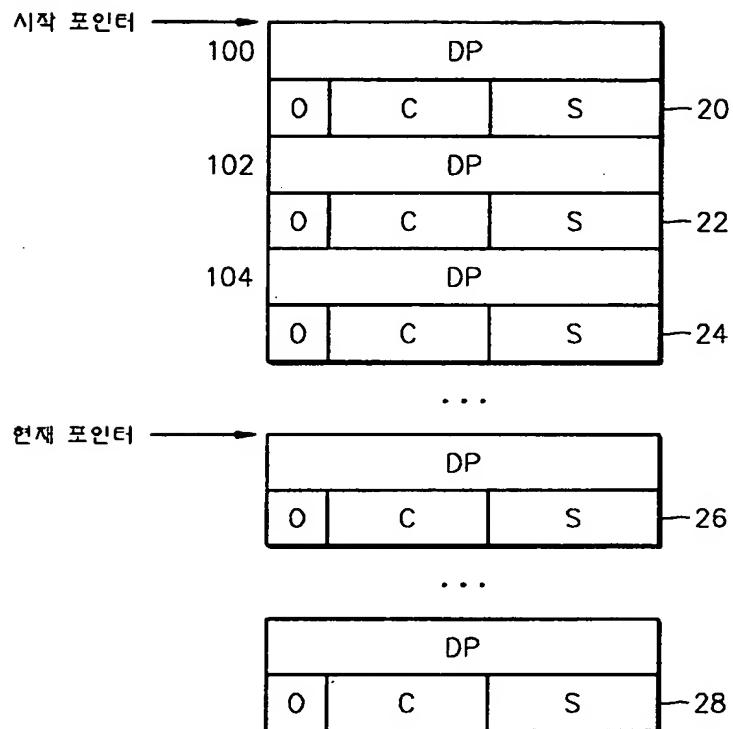
상기 어드레스 카운터는 현재 어드레스 카운터의 값에 단위 버퍼 디스크립터의 크기에 해당되는 어드레스를 가산하여 현재 어드레스 카운터의 값을 갱신하는 것을 특징으로 하는 통신 시스템.

【도면】

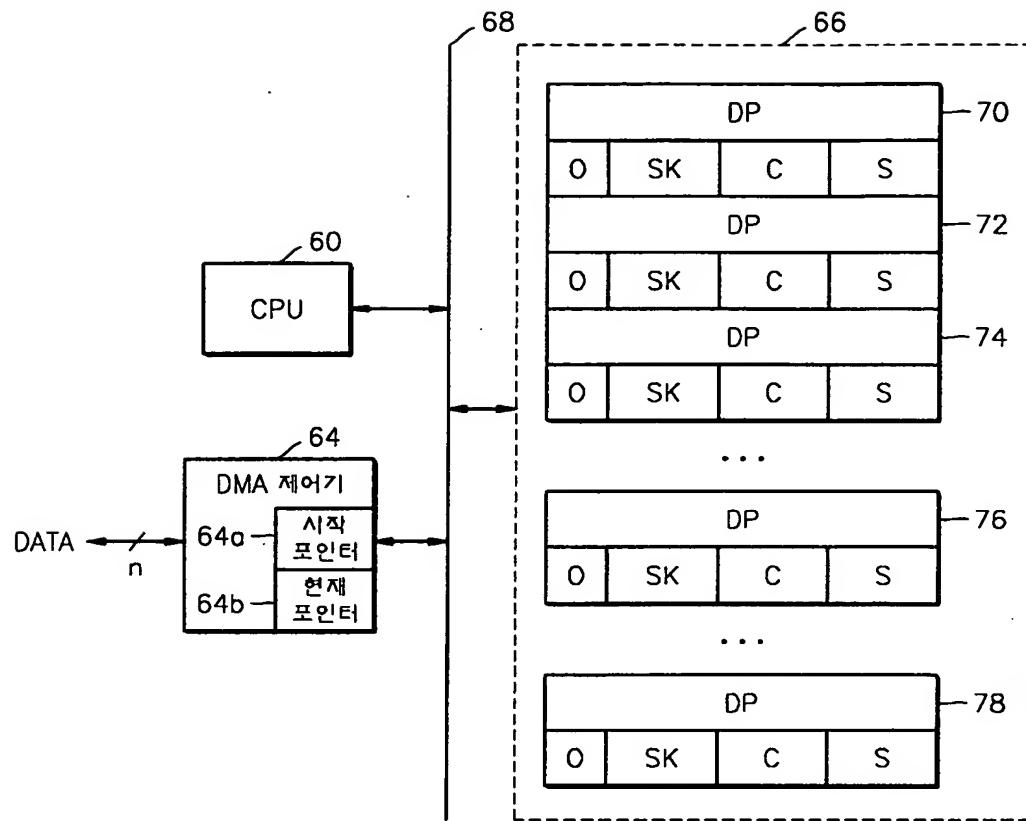
【도 1】



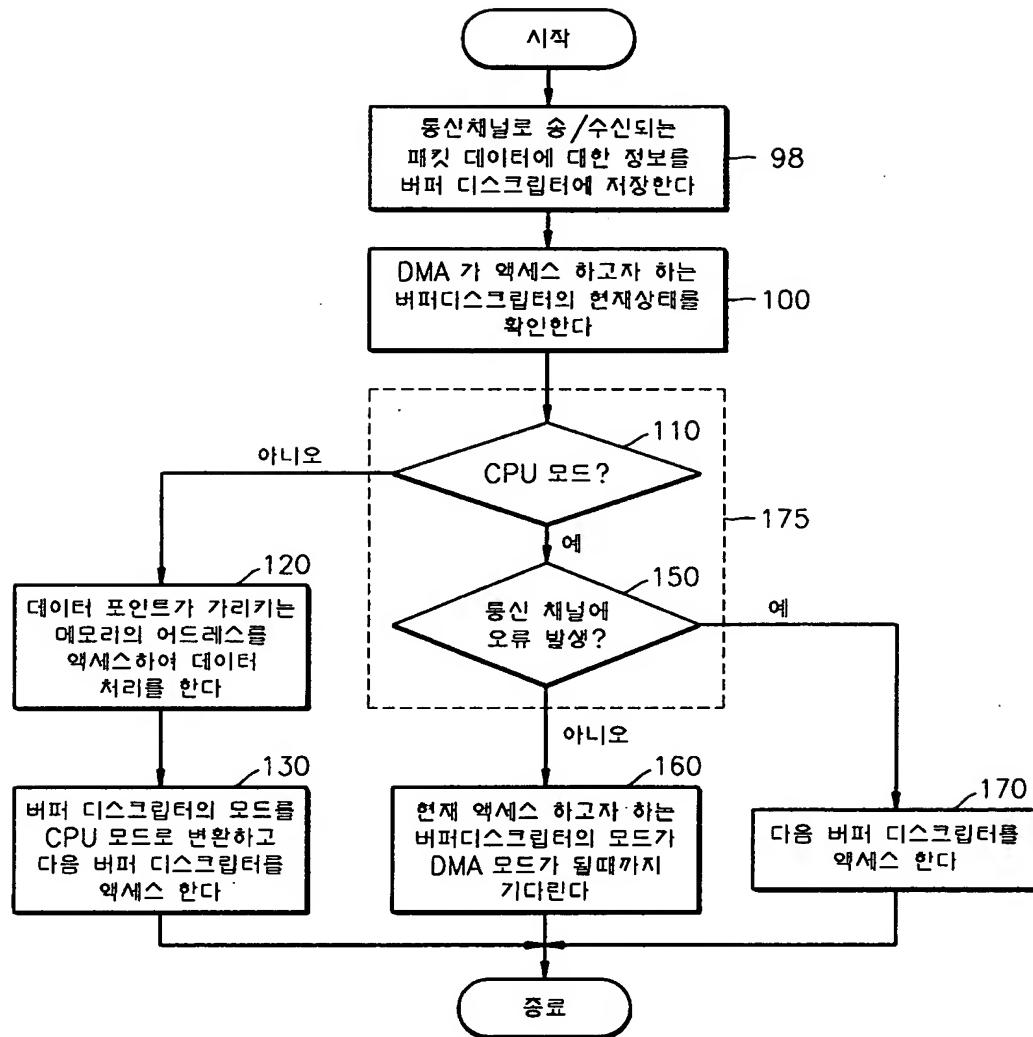
【도 2】



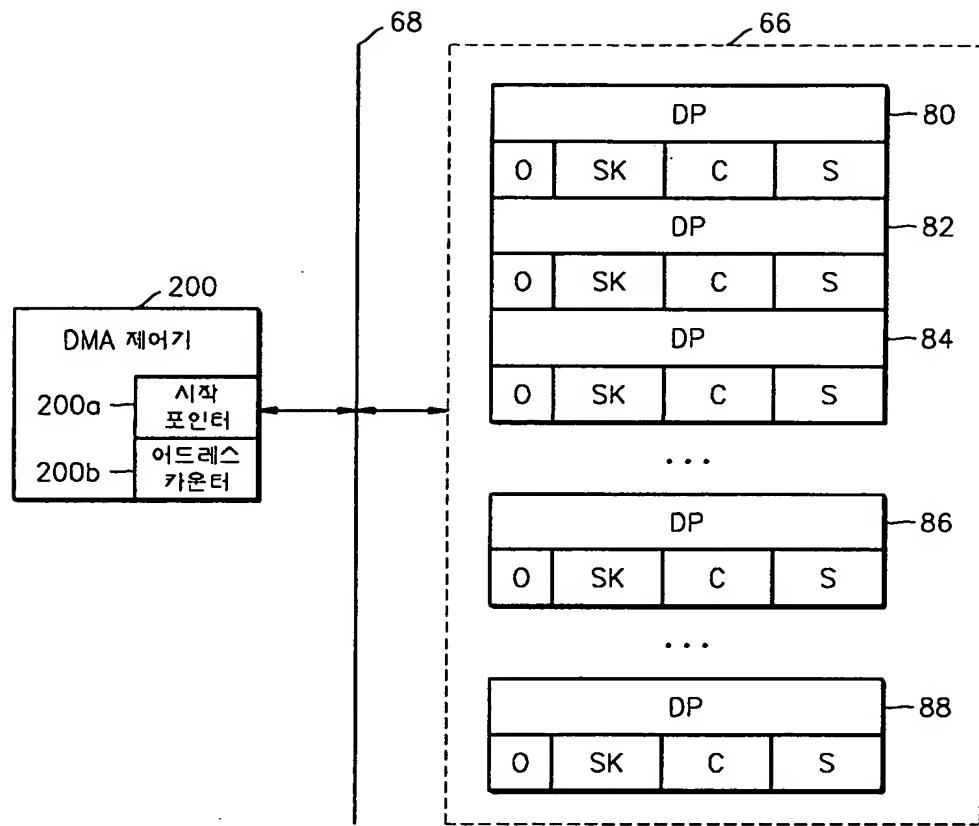
【도 3】



【도 4】



【도 5】



【도 6】

